Information Processing Apparatus and Method

BACKGROUND OF THE INVENTION

Field of the Invention

5   The present invention relates to an information

processing apparatus and method and, more particularly,

to information processing for protecting intellectual

properties (e.g., copyrights) of information data.

Related Background Art

10   Conventionally, as image coding schemes, coding

schemes such as Motion JPEG, Digital Video, and the

like which use intra-frame coding, and coding schemes

H.261, H.263, MPEG-1, MPEG-2, and the like which use

inter-frame predictive coding are known. These coding

15   schemes have been internationally standardized by ISO

(International Organization for Standardization) and

ITU (International Telecommunication Union).

Intra-frame coding is best suited to apparatuses which

require edit and special playback processes of moving

20   images, since it encodes in units of frames, and allows

easy management of frames. On the other hand,

inter-frame coding can assure high coding efficiency

since it uses inter-frame prediction.

         Furthermore, international standardization of

25   MPEG-4 as versatile next-generation multimedia coding

standards which can be used in many fields such as

computers, broadcast, communications, and the like is

in progress.

As such digital coding standards have prevailed, the contents industry strongly recognizes a problem of intellectual property (e.g., copyright) protection. That is, contents cannot be provided with confidence using standards which cannot sufficiently guarantee copyright protection.

To solve this problem, MPEG-4 can describe data in Systems (ISO 14496-1) as Part 1 to build in an IPMP (Intellectual Property Management & Protection) function so as to implement copyright protection in its standards. Refer to ISO 14496-1 for further details.

Fig. 1 shows an example of the format of MPEG-4 encoded data.

Referring to Fig. 1, BIFS (Binary Format for Scene description) encoded data 1010 is obtained by encoding information such as the composition method of object, synchronization of objects, and the like described in Part 1. IPMP encoded data 1011 describes information that pertains to security of bitstreams. Video encoded data 1012 is an encoding result of image data of moving images. Audio encoded data 1013 is an encoding result of audio data appended to moving images.

Fig. 2 shows an example of information described in the IPMP encoded data 1011.

The data 1011 contains information (IPMP object encoded data) indicating encoded data set with

security. In Fig. 2, the video encoded data 1012

corresponds to such data. Authentication data used to

discriminate if decoding of this encoded data is

permitted follows. In Fig. 2, "nonac" is

5   authentication data. In general, this data is

enciphered. In this example, the authentication data

"nonac" which is provided by reversing authentication

data "canon" is enciphered data.

Furthermore, the numbers of frames to be protected

10   and information obtained by enciphering a decipher key

in case of that frame data are enciphered by scrambling

are described to assure security for some parts of a

sequence of video data. In Fig. 2, frame Nos. 1 to 100

in video encoded data can be descrambled by decipher

15   key "key", and frame Nos. 1000 to 1260 can be

descrambled by decipher key "maeda". The IPMP encoded

data 1011 is obtained by encoding these data. The

video encoded data 1012 is copyrighted by scrambling

frames for which the IPMP encoded data assures

20   security.

Fig. 3 shows an example of a decoding apparatus

for decoding such encoded data.

Referring to Fig. 3, an input terminal 1000

receives the IPMP encoded data 1011 and video encoded

25   data 1012 of the encoded data of moving image data. A

demultiplexer 1001 demultiplexes the input encoded data

into the IPMP encoded data 1011 and video encoded data

1012. A buffer 1002 stores the video encoded data 1012 demultiplexed by the demultiplexer 1001 in units of frames.

An IPMP decoder 1003 decodes the IPMP encoded

5   data. An authentication unit 1004 authenticates decoded data. Selectors 1005 and 1007 change their input source/output destination in accordance with the output from the authentication unit 1004. A descrambler 1006 is connected to the selector 1005. A

10 video decoder 1008 decodes video encoded data to play back image data. An output terminal 1009 outputs playback image data.

The operation of the decoding apparatus with the above arrangement will be explained below.

15 The IPMP encoded data 1011 is input first from the input terminal 1000. The demultiplexer 1001 inputs the IPMP encoded data to the IPMP decoder 1003. The IPMP decoder 1003 decodes the IPMP encoded data 1011 to obtain authentication data, frame numbers as security

20 objects, and keys for descrambling.

The authentication data is input to the authentication unit 1004 and is compared with authentication data registered in advance. If authentication is unsuccessful, the selector 1005 is

25 directly connected to the selector 1007 not to output decoded data via the descrambler 1006. If authentication is successful, whether or not the

selectors 1005 and 1007 are connected via the descrambler 1006 is selected in accordance with an instruction from the IPMP decoder 1003.

In this case, when the IPMP decoder 1003 recognizes encoded data of a frame to which security is given, it instructs the selectors 1005 and 1007 to select a path via the descrambler 1006. Otherwise, the decoder 1003 instructs the selectors 1005 and 1007 to select a path without the intervention of the descrambler 1006.

That is, the selectors 1005 and 1007 select each other as the output and input when authentication is unsuccessful in the authentication unit 1004 and when authentication is successful and the IPMP decoder 1003 determines that the descrambler 1006 is not required (i.e., no processing of the descrambler 1006 is done). On the other hand, when authentication is successful, and the IPMP decoder 1003 recognizes encoded data of a frame to which security is given, the selectors 1005 and 1007 select a path via the descrambler 1006.

Therefore, when authentication is successful, the video decoder can play back a normal image since the descrambler descrambles frames to which security is given in addition to those to which no security is given. When authentication is unsuccessful, since scrambled encoded data is input to the video decoder 1008 which cannot normally decode it, no normal

playback image is generated.

However, in this arrangement, IPMP encoded data must be edited upon editing video data, resulting in complicated processes. For example, when a single

5 bitstream is to be generated by combining with another sequence, the frame numbers change, and the contents of the IPMP encoded data must be changed. Key information for scrambling must be prepared independently of security object data, and redundant information must be

10 appended.

SUMMARY OF THE INVENTION

It is, therefore, one object of the present invention to provide an information processing

15 apparatus and method, which can efficiently protect intellectual properties (e.g., copyrights) of information data in consideration of the aforementioned situation.

In order to achieve the above object, an

20 information processing apparatus/method according to one preferred aspect of the present invention is characterized by inputting information data, generating security data to be used to protect the information data, encoding the information data to generate encoded

25 data, extracting a unique predetermined code indicating a specific meaning from encoded data within a security section in accordance with the security data,

superimposing the security data on the predetermined code, scrambling the encoded data except for the predetermined code within the security section, and outputting the superimposed predetermined code and the scrambled encoded data.

An information processing apparatus/method according to another preferred aspect of the present invention is characterized by inputting encoded data in which security data is adaptively superimposed on a unique predetermined code in the encoded data, which indicates a specific meaning, and the encoded data except for the predetermined code is adaptively scrambled in accordance with the security data, extracting from the encoded data a code which is located at a position where the predetermined code is present, detecting the security data from the extracted code, and descrambling the encoded data in accordance with the detection result.

An information processing method according to another preferred aspect of the present invention is characterized by inputting image encoded data that forms a hierarchical structure, extracting a predetermined code indicating a head of a predetermined layer from the image encoded data, and superimposing security data for image protection onto the predetermined code extracted in the extraction step.

An information processing method according to

another preferred aspect of the present invention is characterized by inputting encoded data in which security data is superimposed on a predetermined code indicating a head of a predetermined layer of image

5 encoded data that forms a hierarchical structure, extracting from the encoded data a code which is located at a position where the predetermined code is present, detecting the security data from the extracted code, and decoding the encoded data in accordance with

10 the detection result.

Other objects, features and advantages of the invention will become apparent from the following detailed description taken in conjunction with the accompanying drawings.

15

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 shows an example of the format of MPEG-4 encoded data;

Fig. 2 shows an example of information contained

20 in IPMP encoded data;

Fig. 3 is a block diagram showing the arrangement of a conventional decoding apparatus;

Fig. 4 is a block diagram showing the arrangement of an information processing apparatus according to the

25 first embodiment of the present invention;

Fig. 5 is a block diagram showing the arrangement of an information processing apparatus according to the

second embodiment of the present invention;

Fig. 6 is a block diagram showing the arrangement of an information processing apparatus according to the third embodiment of the present invention;

5   Fig. 7 is a block diagram showing the arrangement of an information processing apparatus according to the fourth embodiment of the present invention;

Fig. 8 is a block diagram showing the arrangement of an information processing apparatus according to the

10   fifth and sixth embodiments of the present invention;

Fig. 9 is a flow chart showing an image encoding process in the fifth embodiment of the present invention;

Fig. 10 is a flow chart showing an image encoding

15   process in the fifth embodiment of the present invention;

Fig. 11 is a flow chart showing an image decoding process in the sixth embodiment of the present invention; and

20   Fig. 12 is a flow chart showing an image decoding process in the sixth embodiment of the present invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

25   Preferred embodiments of the present invention will now be described in detail with reference to the accompanying drawings.

Fig. 4 is a block diagram showing the arrangement of an information processing apparatus according to the first embodiment of the present invention. Note that this embodiment will exemplify a process for

5   copyrighting image data, but the present invention is not limited to image data but may be applied to audio data and the like. As a video coding scheme, MPEG-4 will be exemplified. However, the present invention is not limited to such specific scheme, and other coding

10 schemes such as H.261, H.263, MPEG-1, MPEG-2, and the like may be used.

Referring to Fig. 4, an input terminal 100 inputs image data in units of frames. A video encoder 101 encodes moving images by MPEG-4. A security setting

15 unit 104 sets security for protecting (e.g., copyrights) of moving images. Selectors 102 and 112 select their input source/output destination in accordance with an instruction from the security setting unit 104.

20 A buffer 103 temporarily stores encoded data. A detector 105 detects a start code from the encoded data. A key generation unit 108 generates a key for scrambling encoded data to protect security. An authentication setting unit 110 sets authentication

25 data required for canceling security.

An enciphering unit 109 enciphers input data by a given encipherment scheme, and outputs enciphered data.

A cipher superimposition unit 106 superimposes enciphered data on the start code. A scrambler 107 scrambles in accordance with the aforementioned key. A compositing unit 111 composites the outputs from the

5 cipher superimposition unit 106 and descrambler 107. An input terminal 113 externally inputs audio encoded data.

A multiplexer 114 multiplexes audio data in units of frames of video data. An output terminal 115

10 outputs generated encoded data.

The operation of an image data processing apparatus with the above arrangement will be described below.

The user determines and sets a section of image

15 data to be copyrighted using the security setting unit 104. In this embodiment, security is given to frame Nos. n to (n+m) in a sequence of image data that starts from frame No. 1.

Image data is input from the input terminal 100 in

20 units of frames, and is encoded by the video encoder 101. The encoded data is input to the selector 102 in units of frames. In this case, a frame (frame Nos. 1 to (n-1), or frame Nos. (n+m+1) or later) to which no security is given by the security setting unit 104 is

25 input, the buffer 103 is set as the output of the selector 102 and the input of the selector 112. In this case, encoded data is output from the output

terminal 115 without any security (e.g., scrambling) process.

A process for a frame (frame Nos. n to (n+m)) to which the security setting unit 104 gives security will be explained below. The security setting unit 104 makes the key generation unit 108 generate a key used to scramble the corresponding frame. The generated key is input to the scrambler 107 and enciphering unit 109.

The authentication setting unit 110 sets data required for authentication, e.g., a password, and inputs it to the enciphering unit 109. The enciphering unit 109 enciphers them by a predetermined encipherment scheme to generate enciphered data of 32 bits or less. In this embodiment, 23-bit enciphered data α is generated.

The selector 102 selects the start code detector 105 as its output destination, and the selector 112 selects the compositing unit 111 as its input source. Therefore, the encoded data input to the selector 102 is input to the start code detector 105.

Table 1 below summarizes start codes used in MPEG-4.

Table 1

Code Name	Code (hex)
video_object_start_code	00000100 to 0000011F
video_object_layer_start_code	00000120 to 0000012F
Reserved	00000130 to 000001AF
visual_object_sequence_start_code	000001B0
visual_object_sequence_end_code	000001B1
user_data_start_code	000001B2
group_of_vop_start_code	000001B3
video_session_error_code	000001B4
visual_object_start_code	000001B5
vop_start_code	000001B6
Reserved	000001B7 to 000001B9
face_object_start_code	000001BA
face_object_plane_start_code	000001BB
mesh_object_start_code	000001BC
mesh_object_plane_start_code	000001BD
still_texture_object_start_code	000001BE
texture_spatial_layer_start_code	000001BF
texture_snr_layer_start_code	000001C0
Reserved	000001C1 to C5
System start codes (see note)	000001C6 to 000001FF

In this embodiment, the start code detector 105 detects a VOP (Video Object Plane) start code (vop_start_code). The encoded data other than this

start code is input to the scrambler 107.

The VOP start code detected by the start code detector 105 is input to the cipher superimposition unit 106. The cipher superimposition unit 106 shifts

5 enciphered data α leftward by 9 bits, and performs an exclusive OR operation on the shifted data and VOP start code. For example, if enciphered data α is "7234A" (hexadecimal), the output from the cipher superimposition unit 106 is "E4715B6".

10 The key generated by the key generation unit 108 is input to the scrambler 107, which scrambles the encoded data except for the start code using the key, and inputs the scrambled data to the compositing unit 111. The compositing unit 111 composites data obtained

15 by superimposing the enciphered data α on the VOP start code with the scrambled encoded data to have the former data at the head position.

Audio encoded data encoded by an external circuit is input from the input terminal 113, and is

20 multiplexed with the video encoded data output from the selector 112 in units of frame. The multiplexed data is packetized, and packet data are output from the output terminal 115.

With a series of encoding, enciphering, and

25 selection processes, encoded data security of which is warranted (its copyright is protected) can be generated without any redundancy.

In this embodiment, the start code is selected as a code to be detected. However, the present invention is not limited to such specific code, and any other codes may be detected as long as they are unique codes (indicating specific meanings), even though they have a fixed or variable length from the beginning of the data. In this embodiment, the VOP start code is detected as the start code to be detected, but the present invention is not limited to such specific start code, and higher-order start codes may be detected. For example, the key and authentication information may be superimposed on a VOL start code when security control is made in units of layers, or on a VOS start code when control is made in units of sequences.

Fig. 5 is a block diagram showing the arrangement of an information processing apparatus according to the second embodiment of the present invention.

An input terminal 200 inputs encoded data. This embodiment decodes encoded data generated by the first embodiment, and the input terminal 200 is equivalently connected to the output terminal 115 in Fig. 4.

A demultiplexer 201 demultiplexes multiplexed data into audio encoded data and video encoded data. The video encoded data is output to the subsequent circuits in units of packets.

An output terminal 202 outputs the audio encoded data to an external audio decoder. A start code

detector 203 detects encoded data β located at the position of the VOP start code from the input video encoded data. A buffer 204 stores encoded data other than that corresponding to the position of the start

5 code. An error analyzer 205 extracts a difference from the VOP start code when a normal VOP start code cannot be obtained from encoded data β. A deciphering unit 206 deciphers enciphered data by a predetermined scheme. An authentication unit 207 compares

10 authentication data extracted from the encoded data with input authentication data, and determines to cancel security when they match. An authentication memory 208 stores authentication data unique to this information processing apparatus.

15 Selectors 209 and 211 select their input/output in accordance with instructions from the authentication unit 207 and error analyzer 205. A descrambler 210 descrambles scrambled data. A video decoder 212 decodes video encoded data. An output terminal 213

20 outputs the decoded playback image data.

The operation of an image data processing apparatus with the above arrangement will be described below.

Encoded data input from the input terminal 200 is

25 input to the demultiplexer 201. The demultiplexer 201 demultiplexes the encoded data in units of packets into video encoded data and audio encoded data, and

externally outputs the audio encoded data from the
output terminal 202.

The video encoded data is input to the start code
detector 203. A VOP start code as a start code to be
5   detected is always located at a 32-bit portion from the
head of data when video data is encoded and packetized
in units of frames. For this reason, encoded data for
32 bits from the head position is extracted, and is
output to the error analyzer 205. The remaining data
10   is output to the buffer 204.

The error analyzer 205 compares the input 32-bit
encoded data with the VOP start code to obtain
difference therebetween. That is, the error analyzer
205 computes the OR of the input data and VOP start
15   code value (000001B6), and shifts the OR rightward by 9
bits. The shift result is input to the deciphering
unit 206.

When the input encoded data and VOP start code are
exactly the same, i.e., when no security is given, the
20   error analyzer 205 instructs to directly connect the
selectors 209 and 211 to input the input encoded data
to the video encoder 212.

When security is given, and when the output from
the authentication unit 207 indicates successful
25   authentication, the selectors 209 and 211 are
controlled to be connected via the descrambler 210;
when authentication is unsuccessful, the selectors 209

and 211 are instructed to be directly connected to input the input encoded data to the video decoder 212.

The deciphering unit 206 deciphers the enciphered data by a predetermined scheme to obtain a scrambling

5   key and authentication data. This deciphering scheme deciphers data enciphered by the enciphering unit 109 of the first embodiment. The key obtained by deciphering is input to the descrambler 210, and authentication data is input to the authentication unit

10 207.

The authentication memory 208 stores authentication data unique to this apparatus. The authentication unit 207 compares the authentication data input from the deciphering unit 206 with the

15 authentication data stored in the authentication memory 208. If authentication is successful, and the error analyzer 205 can detect the presence of enciphered data, the selectors 209 and 211 are instructed to select a process in the descrambler 210; otherwise, the

20 selectors are instructed to be connected directly.

The output from the selector 211 is input to the video decoder 212. The video decoder 212 decodes input data to generate a playback image. As a result, as for frames to which no security is given, the video decoder

25 212 directly receives and decodes the output from the demultiplexer 201 via the buffer 204 and the selectors 209 and 211 and can obtain a normal playback image.

On the other hand, when security is set, if authentication fails, the scrambled video encoded data is input, and since the enciphered data remains superimposed on the VOP start code, the VOP start code

5   cannot be detected. Therefore, the encoded data cannot be decoded, and a normal playback image cannot be obtained. Even when security is set, if authentication is successful, the scrambled data can be descrambled, and a normal playback image can be obtained.

10   With a series of selection, deciphering, and decoding processes, image playback can be done in accordance with the intended security level.

In this embodiment, the output from the error analyzer 205 is input to the deciphering unit 206. In

15   addition, the presence/absence of an error may be checked, and if any error is found, the video decoder 212 may be controlled to stop. In this manner, even when no function of detecting a start code at the head of data is available, the video decoder can be

20   prevented from erroneously operating due to wrong data.

Fig. 6 is a block diagram showing the arrangement of an information processing apparatus according to the third embodiment of the present invention. Note that the same reference numerals denote the same building

25   components as in the first embodiment (Fig. 4) mentioned above, and a detailed description thereof will be omitted.

An authentication setting unit 301 allows the user to set authentication data. An IPMP encoder 302 generates IPMP encoded data. An enciphering unit 303 enciphers in the same manner as the enciphering unit

5   109 of the first embodiment, but enciphers a key alone. A cipher superimposition unit 304 superimposes the enciphered data on a start code as in the cipher superimposition unit 106 of the first embodiment. A multiplexer 305 also multiplexes IPMP encoded data in

10   addition to video and audio encoded data of the multiplexer 114 of the first embodiment.

The operation of the an image data processing apparatus with the above arrangement will be described below.

15   The user determines and sets a section of image data to be copyrighted using the security setting unit 104 as in the first embodiment. The authentication setting unit 301 makes the operator set data required for authentication, e.g., a password, and inputs it to

20   the IPMP encoder 302. The IPMP encoder 302 gives security to (e.g., copyrights) the video encoded data in accordance with the IPMP format, and enciphers and encodes the data required for authentication set by the authentication setting unit 301. The output from the

25   IPMP encoder 302 is multiplexed by the multiplexer 305 prior to the video and audio encoded data, and is output from the output terminal 115.

Image data is input from the input terminal 100 in units of frames, and is encoded by the video encoder 101. The encoded data is input to the selector 102 in units of frames. For image data of frames to which no security is given by the security setting unit 104, no security such as scrambling is given to encoded data as in the first embodiment, and the encoded data is output from the output terminal 115.

A process for image data of frames to which the security setting unit 104 gives security will be explained below.

The security setting unit 104 makes the key generation unit 108 generate a key used to scramble the corresponding frame. The generated key is input to the scrambler 107 and enciphering unit 303. The enciphering unit 109 enciphers the key by a predetermined encipherment scheme to generate enciphered data of 32 bits or less. In this embodiment, 32-bit enciphered data γ is generated.

The selector 102 selects the start code detector 105 as its output destination, and the selector 112 selects the compositing unit 111 as its input source. Therefore, the encoded data input to the selector 102 is input to the start code detector 105. The start code detector 105 detects a VOP start code. Encoded data other than the VOP start code is input to the scrambler 107. The detected VOP start code is input to

the cipher superimposition unit 304. The cipher

superimposition unit 304 performs an exclusive OR

operation on the enciphered data γ and VOP start code.

For example, if the enciphered data γ is "7234A19C"

5   (hexadecimal), the output from the cipher

superimposition unit 304 is "7234A02A".

The key generated by the key generation unit 108

is input to the scrambler 107, which scrambles the

encoded data except for the start code using the key,

10   and inputs the scrambled data to the compositing unit

111. The compositing unit 111 composites data obtained

by superimposing the enciphered data γ on the VOP start

code with the scrambled encoded data to have the former

data at the head position.

15   Audio encoded data encoded by an external circuit

is input from the input terminal 113, and is

multiplexed by the multiplexer 305 with the video

encoded data output from the selector 112 in units of

frame. The multiplexed data is packetized, and packet

20   data are output from the output terminal 115.

With a series of encoding, enciphering, and

selection processes, encoded data security of which is

warranted (its copyright is protected) can be generated

without any redundancy. Since details of security for

25   video are contained in the video encoded data, and the

entire video encoded data is managed by IPMP, the video

data can be easily managed.

Fig. 7 is a block diagram showing the arrangement of an information processing apparatus according to the fourth embodiment of the present invention. Note that the same reference numerals denote the same building

5   components as in the second embodiment (Fig. 5), and a detailed description thereof will be omitted. This embodiment processes encoded data generated by the third embodiment, and the input terminal 200 shown in Fig. 7 is equivalently connected to the output terminal

10   115 in Fig. 6.

Referring to Fig. 7, a demultiplexer 401 demultiplexes multiplexed data into IPMP encoded data, audio encoded data, and video encoded data. The video encoded data is output to the subsequent circuits in

15   units of packets. An IPMP decoder 402 decodes IPMP information.

A terminal station 403 allows the user to input authentication data. An error analyzer 404 extracts the difference between the encoded data and VOP start

20   code extracted by the start code detector 203. A deciphering unit 405 deciphers enciphered data by a predetermined scheme. A video decoder 406 outputs playback image data and data indicating if decoding is successful. A frame memory 407 holds last image data

25   for one frame, which is normally decoded. A selector 408 selects and outputs one of the output from the video decoder 406 and the output from frame memory 407

in accordance with the decoding result of the video
decoder 406.

The operation of an image data processing
apparatus with the above arrangement will be described
5   below.

Encoded data input from the input terminal 200 is
input to the demultiplexer 401. The demultiplexer 401
demultiplexes IPMP encoded data and the encoded data in
units of packets into video encoded data and audio
10   encoded data. The demultiplexer 401 externally outputs
the audio encoded data from the output terminal 202.

The IPMP encoded data is output to the IPMP
decoder 402. The IPMP decoder 402 decodes the IPMP
encoded data to acquire information that pertains to
15   IPMP, i.e., authentication data that pertains to a
sequence in this case. The authentication data is
input to the authentication unit 207, and is compared
with authentication data input by the user at the
terminal station 403.

20   The video encoded data demultiplexed by the
demultiplexer 401 is input to the start code detector
203, which separates encoded data corresponding to a
start code and other encoded data as in the second
embodiment, and outputs the data corresponding to the
25   start code to the error analyzer 404, and other data to
the buffer 204.

The authentication unit 207 compares the

authentication data input from the terminal station 403

with the authentication data input from the IPMP

decoder 402. When authentication is successful, and

when the error analyzer 404 can detect the presence of

5   enciphered data, the selectors 209 and 211 are

instructed to select a process in the descrambler 210;

otherwise, the selectors are instructed to be connected

directly.

The error analyzer 404 compares the input encoded

10   data from the first to thirty second bits with the VOP

start code to obtain difference therebetween. That is,

the error analyzer 404 computes the exclusive OR of the

input data and the VOP start code value (000001B6).

The result is input to the deciphering unit 405.

15   When no security is set, the selectors 209 and 211

are instructed to input the encoded data input by

directly connecting them to the video decoder 406.

When security is set, and when the output from the

authentication unit 207 indicates successful

20   authentication, the input source/output destination of

the selectors 209 and 211 are controlled to output data

that has been processed by the descrambler 210; when

authentication is unsuccessful, the selectors 209 and

211 are instructed to input the encoded data which is

25   input by directly connecting them, to the video decoder

406.

The deciphering unit 405 deciphers enciphered data

by a predetermined scheme to acquire a scramble key.

The deciphered key is input to the descrambler 210.

The output from the selector 211 is input to the video decoder 406. The video decoder 406 decodes input data to generate a playback image. As a result, as for

5 frames to which no security is given, the video decoder 406 directly receives and decodes the output from the demultiplexer 401 via the buffer 204 and the selectors 209 and 211 and can obtain a normal playback image. The decoded image data is externally output from the

10 output terminal 213 via the selector 408, and is stored in the frame memory 407 at the same time.

On the other hand, when security is set, if authentication fails, the scrambled video encoded data is input, and since the enciphered data remains

15 superimposed on the VOP start code, the VOP start code cannot be detected, and the encoded data cannot be decoded. In this case, the selector 408 reads out normally decoded last frame data from the frame memory 407 and outputs it to the terminal 213. At this time,

20 the contents of the frame memory 407 are not updated.

Even when security is set, if authentication is successful, the scrambled data can be descrambled, and a normal playback image can be obtained. In this case as well, the image data is externally output from the

25 output terminal 213 via the selector 408, and is stored in the frame memory 407 at the same time.

With a series of selection, deciphering, and

decoding processes, image playback can be done in accordance with the intended security level. Since information that pertains to scrambling of an image is superimposed on video encoded data, easy management in units of frames is allowed. Furthermore, even when decoding is disabled, since a normally played-back image is output, an image can be prevented from suddenly disappearing and no noise image is displayed, thus preventing the user from being disrupted.

Fig. 8 is a block diagram showing the arrangement of an information processing apparatus according to the fifth embodiment of the present invention. In this embodiment, an encoding process of image data will be particularly explained. This embodiment will exemplify MPEG-1, but the present invention is not limited to such specific scheme. Refer to ISO 11172-2 for detailed specifications of MPEG-1.

Referring to Fig. 8, a central processing unit (CPU) 500 controls the overall apparatus, and executes various processes. A memory 501 stores an operating system (OS) and software required to control the apparatus, and provides a storage area necessary for arithmetic operations. Note that the memory 501 has an image area which stores the OS for controlling the overall apparatus to run various software programs, and software programs to run, and loads image data to encode it, a code area for temporarily storing encoded

data, and a working area for storing parameters of various arithmetic operations, and the like.

A bus 502 connects various devices to exchange data and control signals. A storage device 503 stores software. A storage device 504 stores moving image data. A monitor 505 displays an image. A communication circuit 508 comprises a LAN, public line, radio line, broadcast wave, or the like. A communication interface 507 sends encoded data to the communication circuit 508. A terminal station 506 is used to start the apparatus and to set security.

In the aforementioned arrangement, prior to processing, the user selects at the terminal station 506 moving image data to be encoded from those stored in the storage device 504, and starts the apparatus. Then, software stored in the storage device 503 is mapped on the memory 501 via the bus 502, and is launched.

The encoding operation of moving image data stored in the storage device 504 by the CPU 500 will be described below with reference to the flow charts shown in Figs. 9 and 10.

The encoding process will be explained below using Fig. 9.

Referring to Fig. 9, security setting conditions such as "sequence authentication data" used to authenticate in units of sequences, "scene

authentication data" used to authenticate in units of sets of scenes, a picture to which security is given to start inhibition of decoding, a picture for which inhibition is canceled, "picture authentication data" used to authenticate in units of pictures, and the like are input in step S1, and are stored in the working area of the memory 501. The flow then advances to step S2.

In step S2, "sequence authentication data" is enciphered and the enciphered data is superimposed on a sequence header code (value "000001B3") of MPEG-1 by an exclusive OR operation, and the result is stored in a predetermined area of the memory 501. The flow advances to step S3.

In step S3, encoded data of another sequence layer is generated, is stored to follow the superimposed sequence header code, and is stored as packets in a predetermined area of the storage device 504. The flow then advances to step S4.

It is checked in step S4 if the process is complete for image data of all scenes to be encoded, when a GOP is considered as one scene. If the process is complete for all image data, the software ends; otherwise, the flow advances to step S5.

In step S5, "GOP authentication data" is enciphered and is superimposed on a GOP start code (value "000001B7") of MPEG-1 by an exclusive OR

operation. The OR is stored in a predetermined area of the memory 501. The flow advances to step S6.

In step S6, encoded data of another GOP layer is generated, is stored to follow the superimposed GOP start code, and is stored as packets in a predetermined area of the storage device 504. The flow advances to step S7.

It is checked in step S7 if the encoding process is complete for all picture image data to be encoded in the GOP. If the encoding process is completed for all image data, the encoding process of the GOP ends, and the flow returns to step S4 to execute the encoding process of the next GOP. Otherwise, the flow advances to step S8 to encode in units of pictures.

The encoding process in units of pictures in step S8 in Fig. 9 will be described below using Fig. 10.

In Fig. 10, it is checked in step S10 based on the picture conditions stored in the working area of the memory 501 in step S1 in Fig. 9 if picture image data to be encoded falls within a security section. If the picture to be encoded is included within the security section, the flow advances to step S12 to encipher it; otherwise, the flow advances to step S11 to execute a normal picture encoding process.

In step S11, image data of the picture to be encoded is read out from the storage device 504, and undergoes picture layer encoding of MPEG-1. The

encoded data is stored as packets in a predetermined area of the storage device 504. The flow then advances to step S7 in Fig. 9 to process the next picture.

In step S12, a key for scrambling is generated,

5 the generated key and "picture authentication data" are enciphered, the enciphered data is superimposed on a picture start code (value "00000100") of MPEG-1 by an exclusive OR operation, and the obtained data is stored in a predetermined area of the memory 501. The flow

10 then advances to step S13.

In step S13, encoded data that pertains to the header of another picture layer is generated, and is stored following the superimposed picture start code. Furthermore, image data of the picture to be encoded is

15 read out from the storage device 504, and is temporarily stored in the code area of the memory 501 after it is encoded.

In step S14, the encoded data generated in step S13 is scrambled using the key generated in step S12,

20 and the flow advances to step S15.

In step S15, the superimposed picture start code and scrambled encoded data are stored as packets in a predetermined area of the storage device 504, and the flow advances to step S7.

25 With a series of encoding, enciphering, and selection processes described above, encoded data for which security is assured can be generated without any

redundant data.   Since security can be set in units of
layers, various security levels can be given.

In the above embodiment, data is stored in the
storage device 504.   Alternatively, the data may be
5   output onto the communication line 506 via the
communication interface 505.

The sixth embodiment will especially explain the
decoding process of image data.   The arrangement of the
information processing apparatus is the same as that
10   shown in Fig. 8 of the fifth embodiment.   Note that
this embodiment also exemplifies MPEG-1, but the
present invention is not limited to this.   In this
embodiment, the decoding process of encoded data
generated in the fifth embodiment and stored in the
15   storage device 504 will be exemplified.

In the arrangement shown in Fig. 8, prior to the
processing, when the user selects at the terminal
station 506 encoded data to be decoded from moving
image encoded data stored in the storage device 504 and
20   starts the apparatus, the software stored in the
storage device 503 is mapped on the memory 501 via the
bus 502 and is launched.

The decoding operation of encoded data stored in
the storage device 504 by the CPU 500 will be explained
25   below using the flow charts shown in Figs. 11 and 12.

The decoding process will be explained below using
Fig. 11.

Referring to Fig. 11, the first packet of the sequence to be decoded is read out from the storage device 504, and is stored in the code area of the memory 501 in step S20. The first 32 bits of the stored encoded data are compared with the sequence header code to demultiplex and decipher the superimposed "sequence authentication data". The flow then advances to step S21.

In step S21, the deciphered "sequence authentication data" is compared with authentication data input from the terminal station 506. If authentication is successful, the flow advances to step S22 to proceed with the decoding process. If authentication is unsuccessful, the decoding process ends to stop software.

In step S22, encoded data of another sequence layer stored in the code layer of the memory 501 is decoded, and the decoding result is stored in the working area of the memory 501, so that the data can be used in the subsequent processes. The flow then advances to step S23.

It is checked in step S23 if the process is complete for image data of all scenes (GOPs). If the process is complete for all image data, the software ends. Otherwise, the flow advances to step S24.

In step S24, a packet associated with the header of the GOP to be decoded is read out from the storage

device 504, and is stored in the code area of the memory 501. The first 32 bits of the stored encoded data are compared with a GOP start code to demultiplex and decipher the superimposed "GOP authentication

5   data", and the flow then advances to step S25.

In step S25, the deciphered "GOP authentication data" is compared with authentication data input from the terminal station 506. If authentication is successful, the flow advances to step S26 to proceed

10   with decoding. If authentication is unsuccessful, the flow returns to step S23 to try to process the next GOP.

In step S26, encoded data of another GOP layer stored in the code area of the memory 501 is decoded,

15   and the decoding result is stored in the working area of the memory 501, so that the data can be used by the subsequent processes. The flow then advances to step S27.

It is checked in step S27 if the process is

20   complete for all picture image data to be decoded in the GOP. If the decoding process is complete for all image data, the decoding process of the GOP ends, and the flow returns to step S23 to decode the next GOP. Otherwise, the flow advances to step S28 to decode in

25   units of pictures.

The decoding process in units of pictures in step S28 in Fig. 11 will be described below using Fig. 12.

In Fig. 12, packets associated with the picture to be decoded are read out from the storage device 504 and are stored in the code area of the memory 501 in step S31.

5   In step S32, the first 32 bits of the encoded data stored in the memory 501 are compared with a picture start code. If the value is "00000100", the flow advances to step S33 to execute a normal picture decoding process. That is, no security is set. On the

10   other hand, if the value is not "00000100", since security is set, the flow advances to step S34 to process that data.

In step S33, encoded data of the picture to be decoded is read out from the code area of the memory

15   501 and undergoes picture layer decoding of MPEG-1. The decoded data is sent to and displayed on the monitor 505. To process the next picture, the flow advances to step S27 in Fig. 11.

On the other hand, in step S34 the superimposed

20   "picture authentication data" and key for scrambling are demultiplexed and deciphered, and the flow advances to step S35.

In step S35, the deciphered "picture authentication data" is compared with authentication

25   data input from the terminal station 506. If authentication is successful, the flow advances to step S36 to proceed with the decoding process. If

authentication is unsuccessful, the flow advances to step S27 in Fig. 11 to process the next picture.

In step S36, encoded data of the picture layer stored in the code area of the memory 501 is read out, and is descrambled by the deciphered key. The flow advances to step S37.

In step S37, the descrambled picture encoded data undergoes picture layer decoding of MPEG-1, and the decoded data is sent to and displayed on the monitor 505. The flow then advances to step S27 in Fig. 11 to process the next picture.

With a series of selection, deciphering, and decoding processes, image playback can be done in accordance with the intended security level.

Since security can be set in units of layers, various security levels can be set. Since a system, which cannot decipher each authentication data, cannot recognize each start code, it cannot play back any data, thus implementing copyright protection.

As can be seen from the above description, in this embodiment, encoded data to which security is given while suppressing redundant data to be appended can be generated.

In this embodiment, security data can be set in correspondence with layers. Since security can be managed in units of pictures or frames, the present invention can be suitably applied to an edit process

and the like.

In other words, the foregoing description of embodiments has been given for illustrative purposes only and not to be construed as imposing any limitation 5 in every respect.

The scope of the invention is, therefore, to be determined solely by the following claims and not limited by the text of the specifications and alterations made within a scope equivalent to the scope 10 of the claims fall within the true spirit and scope of the invention.